International Journal of Engineering Research and Science & Technology

IJERST

ISSN : 2319-5991

www.ijerst.com

Email: editor@ijerst.com or editor.ijerst@gmail.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

UPI FRAUD DETECTION USING MACHINE LEARNING ALGORITHMS

Jallapuram Sindhu, Ms. Vijaya Sree Swarupa

UG Student, Department of Electronics and Computer Engineering, JBIET, India.

Assistant professor, Department of Electronics and Computer Engineering, JBIET, India.

ABSTARCT

Increase in UPI usage for online payments, Cases of fraud associated with it are also rising. Few steps involving UPI transaction process using a Hidden Markov Model (HMM). An HMM is initially trained for a cardholder. If a UPI transaction is not accepted by the trained HMM. It is considered to be fraudulent. People can use UPIs for online transactions as it provides an efficient and easy-to-use facility. With the increase in usage of UPIs, the capacity of UPI misuse has also enhanced. UPI frauds cause significant financial loses for both UPI holders and financial companies. In this project, the main aim is to detect such frauds, including the accessibility of public data, high-class imbalance data, the changes in fraud nature, and high rates of false alarm. The main focus has been to apply the recent development of machine learning algorithms for this purpose. We have created 5 Algorithms to detection the UPI Fraud and evaluated results based on that. Various modern techniques like artificial neural network. Different machine learning algorithms are compared, including Auto Encoder, Local Outlier Factor, Kmeans Clustering. This project uses various algorithms, and neural network which comprises of techniques for finding optimal solution for the problem and implicitly generating the result of the fraudulent transaction. This algorithm is a heuristic approach used to solve high complexity computational problems. The implementation of an efficient fraud detection system is imperative for all UPI issuing companies and their clients to minimize their losses.

Literature review

As digital payments grow in popularity, UPI transactions have seen widespread adoption, but this has also led to a rise in fraud cases. Studies and reports highlight several categories of UPI fraud, including phishing scams, social engineering attacks, and man-in-the-middle (MITM) attacks. Research shows that the increased reliance on mobile applications for financial transactions introduces new security risks, especially in markets where smartphone usage is high but digital literacy may vary significantly. For example, a report by the Reserve Bank of India in recent years has noted a substantial increase in complaints related to unauthorized UPI transactions, indicating the need for more robust security and fraud detection mechanisms.

Understanding UPI Transactions

The Unified Payments Interface (UPI) system, developed by the National Payments Corporation of India (NPCI), provides a seamless way to transfer funds instantly between two bank accounts. Each UPI transaction involves several key elements, including:

Payer (Sender) and Payee (Receiver): The sender initiates the transaction to transfer funds to the receiver.

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

UPI ID/Virtual Payment Address (VPA): Unique identifiers linked to user's bank accounts, used in place of numbers.

Bank and NPCI Servers: Intermediary systems that facilitate authentication and transfer requests.

Two-Factor Authentication (2FA): A UPI mandate for security, where users verify their identity using a PIN, usually entered through a mobile device.

Transaction Timestamps: Date and time stamps, which help identify the exact moment each transaction occurs.

The UPI transaction flow begins with the sender initiating a request to transfer a specific amount. After authorization via 2FA, the transaction request is routed through the NPCI servers and the respective banks. Once verified, the funds are transferred instantly, with a notification sent to both parties confirming the successful transaction. This rapid and straightforward process makes UPI attractive but also opens avenues for exploitation by fraudsters who manipulate each element of the transaction flow.

Methodology

The methodology section outlines the approach for building an effective UPI fraud detection model. It covers the data collection and preprocessing techniques, feature engineering, algorithm selection, and model building processes used in your project.

Data Collection:

The accuracy and reliability of a fraud detection model rely heavily on the quality of data it's trained on. In UPI fraud detection, data collection involves acquiring historical transaction data with labels indicating whether a transaction was fraudulent or legitimate.

Experimentation and Results

The experimentation and results section documents the process of testing various machine learning models for UPI fraud detection, along with the outcomes and insights gained from evaluating the model's performance. This section highlights the steps taken to test different models, tune them, and compare their effectiveness in identifying fraudulent transactions.

Experimental Setup:

Environment and Tools: The experimentation was conducted using a data science environment such as Jupyter Notebook, with programming in python. Key libraries included **Pandas** for data handling, **Scikit-Learn** for machine learning models, **XGBoost** and **LightGBM** for boosting algorithms, and **Matplotlib** and **Seaborn** for Results and Analysis:

Comparison of Models: The supervised models,, particularly XGBoost and LightGBM, achieved the highest accuracy and recall, indicating they could capture intricate patterns in the transaction data. Autoencoders showed promising results as an anomaly detection method, especially in Identifying unexpected patterns, but

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

required more tuning for consistent results. Hybrid models demonstrated versatility by combining strengths from supervised and unsupervised methods, especially in adapting to new fraud patterns.

Fraud Detection System Design

The design of the UPI fraud detection system involves creating a robust, scalable, and real-time solution to monitor, detect, and mitigate fraud in digital payments. The system's architecture is built to manage high volumes of transaction data, process it efficiently, and flag suspicious activities based on patterns and machine learning models. This section covers the components, data flow, deployment considerations, and critical aspects required for an effective fraud detection system.

Discussion

The discussion section reflects on the findings and implications of implementing a UPI fraud detection system using machine learning. This part evaluates the overall effectiveness, explores observed fraud patterns, assesses model performance, discusses challenges encountered, and considers the broader impact on the digital payments landscape.

Key Findings and Insights:

The UPI fraud detection project reveals several important insights regarding fraud detection techniques and transaction behavior:

- **Patterns in Fraudulent Transactions**: The system uncovered patterns commonly associated with fraudulent transactions, such as unusually high transaction frequency, changes in transaction location, or atypical amounts for specific users. These patterns are essential for developing feature sets and refining the model.
- **Effectiveness of Machine Learning**: Machine learning proved effective for this application, with the model able to recognize complex patterns that traditional rule-based systems might miss. The machine learning approach allowed for improved adaptability as the system learns from new data over time.
- **Feature Importance**: Feature engineering emerged as a critical factor in model performance. Features like transaction frequency, amount variance, and geolocation played significant roles in identifying suspicious transactions. This emphasizes the importance of domain- specific feature selection in fraud detection.
- **System Performance**: The system demonstrated the ability to process high transaction volumes in near-real-time, confirming that a well- optimized machine learning model can handle the speed and scale required for UPI fraud detection.

Model Effectiveness and Accuracy:

- The fraud detection model's accuracy and reliability were primary measures of its effectiveness. Various metrics, including precision, recall, F1-score, and ROC-AUC, indicated the model's robustness in distinguishing legitimate transactions from fraudulent ones. Key takeaways on performance include:

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

- **High Recall, Balanced Precision**: A high recall rate was prioritized to capture as many fraudulent transactions as possible, even at the cost of some false positives. This approach was necessary to ensure that no significant fraudulent activities were missed.

- **Precision Considerations**: A balance was maintained in precision to reduce false positives, though some genuine transactions were occasionally flagged. This balance between precision and recall was critical to minimize interruptions for legitimate users while ensuring fraud coverage.
- **Improvement Over Rule-Based Systems**: Unlike traditional rule- based systems, which are often rigid and easily circumvented by sophisticated fraud tactics, the machine learning model provided a more dynamic solution capable of adapting to evolving fraud techniques.

Conclusion

The UPI Fraud Detection project set out to address the rising concerns of fraud in digital payments by leveraging machine learning techniques. The system aimed to detect potentially fraudulent transactions in real-time, safeguarding user accounts and supporting the security goals of the UPI (Unified Payments Interface) network. This section reflects on the project's primary achievements, limitations, and implications, while highlighting future opportunities for enhancing digital payment security.

Summary of Objectives and Key Achievements:

The primary objective of this project was to design and implement a fraud detection system that could identify potentially fraudulent UPI transactions based on transaction behavior and user patterns. Key achievements include:

- **Effective Fraud Detection**: The machine learning model was successful in identifying patterns associated with fraudulent transactions. Through feature engineering, the system developed a deep understanding of behaviors, such as transaction frequency, transaction value, and user location, that could signal potential fraud.
- **Real-Time Detection Capability**: A major accomplishment was achieving real-time detection and response capabilities. By designing an architecture that could process transaction data instantly and provide timely alerts, the system demonstrated its potential for real-world application in high-throughput UPI environments.
- **Scalable System Design**: The project successfully developed a scalable and modular architecture. By leveraging cloud-compatible components and frameworks, the system is capable of handling the growing volume of UPI transactions, making it adaptable to the increasing adoption of digital payments.
- **Improved Security and User Trust**: By detecting and mitigating fraudulent activities in real time, the project contributes to enhancing user trust in digital payments, supporting the growth of the UPI ecosystem and improving user confidence in electronic transactions.

Reflections on the Use of Machine Learning in Fraud Detection:

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

Machine learning proved to be a robust solution for UPI fraud detection, surpassing traditional rule-based systems in adaptability and precision. Key reflections include:

- **Adaptability and Responsiveness**: The machine learning model's ability to adapt to new fraud patterns over time enables the system to respond to dynamic fraud techniques, a critical advantage over rule-based systems. This adaptability enhances the overall security of UPI transactions.
- **Enhanced Detection Accuracy**: Through advanced feature engineering and model training, the system achieved a high level of accuracy, with well-balanced precision and recall rates. This reflects the potential of machine learning to distinguish between legitimate and fraudulent transactions with a low margin of error.
- **Data Dependency**: The project underscored the importance of quality data for effective model training. To sustain high detection accuracy, ongoing data collection and model retraining are essential to keep the system updated with the latest fraud patterns.

References

The References section lists the key academic papers, articles, technical documentation, and other resources consulted during the project. These references support the development of the UPI fraud detection system by providing insights into machine learning algorithms, fraud detection methodologies, digital payment security, and more.

Academic Journals and Conference Papers:

Academic journals and conference papers are essential for understanding the theoretical background, algorithmic development, and state-of-the-art techniques in fraud detection and machine learning. Typical references might include:

- Fraud Detection Methodologies:

  Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science*, 17(3), 235-255.

  - This paper provides a comprehensive review of statistical techniques in fraud detection, laying the groundwork for understanding statistical and machine learning approaches.
  - Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. *Artificial Intelligence Review*, 34(4), 271-308.
  - This survey article gives insights into data mining and machine learning techniques applied to fraud detection, which helped shape the feature engineering and model selection process.

- Machine Learning in Fraud Detection:
  - Randhawa, K., Jain, S., & Sharma, M. (2018). Credit card fraud detection

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

using AdaBoost and majority voting. *Journal of Advanced Research in Dynamical and Control Systems*, 10(7), 1419-1426.

- This research provides examples of using ensemble learning methods like AdaBoost, which could be valuable for model experimentation in fraud detection.
- This research provides examples of using ensemble learning methods like AdaBoost, which could be valuable for model experimentation in fraud detection.
- Han, J., Kamber, M., & Pei, J. (2011). Data Mining: *Concepts and Techniques.* Elsevier.
- National Payments Corporation of India (NPCI). (n.d.). *Unified Payments Interface (UPI) Procedural Guidelines*. Retrieved from https://www.npci.org.in
- This document provides essential guidelines on UPI transaction handling, security requirements, and technical standards, which were instrumental in designing the system architecture.

  - Data Privacy and Security Regulations:

- General Data Protection Regulation (GDPR). (2018). *Regulation (EU) 2016/679 of the European Parliament and of the Council*. Available at: https://gdpr.eu
- GDPR guidelines were referenced to ensure that the data handling process adhered to international standards for data privacy, especially regarding anonymization and user consent.

Online Resources and Industry Reports:

Online resources and industry reports provide practical insights into current fraud trends, real-world UPI usage patterns, and recent advancements in fraud detection:

Reports on Fraud in Digital Payments:

- Accenture. (2023). *2023 Global Digital Payment Fraud Insights*. Retrieved from https://www.accenture.com
- This industry report outlines recent trends in digital payment fraud, including types of attacks common in UPI transactions, helping inform feature engineering.
- National Payments Corporation of India (NPCI). (2024). *UPI Statistics and Growth Report*. Available at: https://www.npci.org.in
- The report provides recent transaction data and UPI growth trends, which helped shape the system's scalability and performance requirements.

  - Technical Blogs and Guides:

- Towards Data Science. (2022). *How to Build a Real-Time Fraud Detection System Using Machine Learning*. Retrieved from https://towardsdatascience.com
- This blog post offers practical insights into designing and deploying real-time fraud detection systems, guiding the implementation strategy for the project.

Machine Learning Libraries and Tools:

References to the libraries and tools used in the project help provide a technical foundation for reproducibility. Typical references might include:

Machine Learning Libraries:

- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

- Scikit-learn is the main library used for model training and evaluation in this project.
- Chollet, F., & others. (2015). *Keras*. Retrieved from https://keras.io
    - Keras was used for prototyping and experimenting with neural networks, though not as central as traditional ML models for this specific project.
- Data Processing Libraries:
    - McKinney, W. (2010). Data Structures for Statistical Computing in Python. *Proceedings of the 9th Python in Science Conference*, 51-56.
    - The pandas library, which was referenced for data preprocessing and handling, is foundational in the data preparation pipeline.

Future Research Directions:

References for future exploration, such as papers on advanced machine learning methods, can add depth to the project's scope and potential evolution:

- Federated Learning and Privacy:
    - Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... & Zhao, S. (2019). Advances and Open Problems in Federated Learning. *arXiv preprint arXiv:1912.04977*.
    - This paper discusses federated learning, a promising approach for privacy-preserving machine learning, which could be an enhancement for the UPI fraud detection system.
- Graph-Based Models in Fraud Detection:
    - Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
    - This research on Graph Convolutional Networks (GCNs) offers insight into advanced techniques that could detect fraud through relational data, presenting potential future directions.

2. Appendices

The Appendices section provides additional details and supporting information for the UPI fraud detection project. This section includes data dictionaries, code samples, statistical test results, hyperparameter configurations, and other relevant technical details. Each appendix is designed to enhance understanding, ensuring the reproducibility and transparency of the project.

Appendix A: Data Dictionary and Feature Descriptions:

This appendix provides an in-depth data dictionary for the dataset used in the project. It lists and describes each feature used in the model, along with the data type and sample

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

values. This is useful for understanding how specific attributes contribute to fraud detection.

- **Transaction_ID**: Unique identifier for each transaction (String).
- **User_ID**: Anonymized ID representing the user conducting the transaction (String).
- **Transaction_Amount**: The monetary amount of the transaction (Float).
- **Transaction_Type**: The type of transaction (e.g., transfer, bill payment, etc.) (String).
- **Location**: Approximate location of the transaction (String).
- **Device_ID**: Identifier for the user's device (String).
- **Transaction_Timestamp**: The exact timestamp of the transaction (Datetime).
- **Frequency**: The frequency of transactions made by the user within a specified period (Integer).
- **Average_Transaction_Amount**: The average amount transacted by the user within a specified period (Float).
- **Fraud_Label**: Indicates whether the transaction is fraudulent (Binary; 0 = Not Fraud, 1 = Fraud).

This data dictionary ensures clarity on how each variable functions within the fraud detection model.

Appendix B: Model Development Code and Implementation Details:

This appendix provides code snippets and implementation details of the machine learning models used for fraud detection, such as data preprocessing, feature engineering, model training, and evaluation. Detailed code helps replicate or understand the process for technical audiences.

- Data Preprocessing:
  - Code for handling missing values, outlier treatment, and data normalization.
  - Examples of label encoding or one-hot encoding for categorical variables.
- Feature Engineering:
  - Code snippets for creating new features, such as frequency of transactions, average transaction amount, and device ID usage patterns.

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

- Model Training and Evaluation:
  - Code used to split the dataset into training and testing sets.
  - Implementation details for each machine learning model (e.g., Decision Trees, Random Forest, and XGBoost).
  - Code for cross-validation, hyperparameter tuning, and evaluation metrics like confusion matrix, precision, recall, and F1-score.

Appendix C: Hyperparameter Tuning and Optimization Results:

This appendix outlines the hyperparameter tuning experiments performed to optimize model performance. Detailed tables showing the parameter values tested, evaluation scores, and the selected optimal parameters are included.

- Grid Search for Random Forest:
  - Parameters tested: n_estimators, max_depth, min_samples_split.
  - Results table summarizing each configuration and associated accuracy, precision, and recall scores.
- XGBoost Tuning with Randomized Search:
  - Parameters tested: learning_rate, max_depth, subsample, n_estimators.
  - Summary of the best parameter combination along with its performance metrics.

    This appendix allows readers to understand the optimization process and why certain parameters were chosen for each model.

Appendix D: Statistical Analysis and Validation Tests:

Here, include results of any statistical tests conducted to validate assumptions or performance metrics of the models. This section could include information on tests for data distribution, feature importance, and statistical significance.

- Distribution Analysis:
  - Tests for normality or skewness of transaction amounts or frequencies.
  - Box plots or histograms displaying distribution comparisons between fraudulent and legitimate transactions.
- Feature Importance Analysis:
  - Bar charts or tables showing the importance of various features in

ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

models like Random Forest or XGBoost.

- Detailed explanation of feature importance scores, indicating which factors most influence fraud prediction.

Appendix E: Evaluation Metrics and Performance Charts:

This appendix provides detailed performance metrics and visualizations for model evaluation, enabling a deeper understanding of the system's effectiveness.

- Confusion Matrix and Classification Report:
  - Confusion matrix for each model (Random Forest, Decision Tree, XGBoost) on the test dataset.
  - Classification report summarizing precision, recall, F1- score, and accuracy for each model.
- Receiver Operating Characteristic (ROC) Curve:
  - ROC curve plots for each model, along with Area Under Curve (AUC) scores, illustrating the trade-off between true positive and false positive rates.
- Precision-Recall Curve:
  - Precision-recall curves to assess the model's performance, especially useful in imbalanced datasets where fraud cases are less frequent.

Appendix F: System Design Diagrams and Architecture:

Diagrams illustrating the architecture and design of the fraud detection system, including the data flow, processing steps, and model integration.

- System Architecture Diagram:
  - Diagram showing the end-to-end system flow, from data ingestion to preprocessing, model prediction, and alert generation.
- Data Flow and Processing Pipelines:
  - Flowcharts illustrating data flow across various stages: data collection, preprocessing, feature engineering, model prediction, and logging results.

This appendix clarifies the overall design, making it easier for others to understand and potentially replicate the system setup.

Appendix G: Additional Findings and Observations:

Include additional findings or exploratory insights that were not central to the main

International Journal of Engineering Research and Science & Technology
ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 4, 2024

results but are noteworthy. For instance:

- Pattern Observations in Fraudulent Transactions:
  - Patterns such as unusual transaction timings, device ID mismatches, or locations distant from usual transaction areas.
- Case Studies or Anomalies:
  - Detailed analysis of specific cases flagged by the model, exploring what led to those classifications and the model's reasoning.
    These insights might provide useful context for understanding nuanced fraud patterns.

Formatting Tips for Appendices:

- **Consistency:** Label each appendix clearly (e.g., Appendix A, Appendix B) and include a title and brief description.
- **Referencing**: Reference each appendix within the main text where relevant (e.g., "See Appendix A for the data dictionary").
- **Clarity and Detail**: Ensure that each appendix is organized logically, with explanations and legends for any charts, tables, or code snippets.